Migration Letters

Volume: 21, No: S5 (2024), pp. 339-361

ISSN: 1741-8984 (Print) ISSN: 1741-8992 (Online)

www.migrationletters.com

Optimized Adaptive Fuzzy Expert System-Based Plant Leaf Disease Prediction Model Using Data Through Internet Of Things

Manjiri U. Karande[1] , Shrikant L. Satarkar[2]

Abstract:
Agriculture serves as the fundamental backbone of a nation, accounting for almost 50% of the global economy. Precision agriculture is crucial for assessing the condition of crops in order to identify appropriate measures for plant care. The given text is incomplete and does not provide enough information to rewrite it in a straightforward and precise manner. Please provide more context or complete the sentence. The traditional approach of predicting leaf diseases lacks stability and only offers limited accuracy in its predictions.
This study focuses on creating an enhanced module for predicting leaf illnesses with high accuracy. The module utilizes a hybrid optimization guided adaptive fuzzy expert system for disease detection. The Internet of Things (IoT) is recognized for its ability to gather real-time data. The suggested model makes use of the data acquired via the IoT framework. The data is analyzed to identify the existence of diseases in the crop, facilitated by the suggested Cat swarm-based Harris Hawks (CSHH) optimization method. The CSHH optimization method will be created by integrating the key features of the cat swarm optimization (CSO) algorithm and the Harris Hawks optimization (HHO) algorithm.

Keywords: *IoT, Fuzzy Expert System, CSHH, CSO, HHO.*

1. Introduction

Disease of the plant causes the withering of the flowers, fruit, and foliage, and in extreme cases, plant death, which can result from failure to detect and prevent the disease. Therefore, in order to diagnose plant leaf[1] diseases more precisely, an automatic recognition of diseases mechanism with enhanced precision is required. This necessitates a reevaluation of conventional approaches to plant disease identification mechanisms, including an examination of their accomplishments, challenges, and methods, in order to incorporate them into a novel method that provides more precise predictions[1-5].

The implementation of smart agriculture leverages the potential of the internet of things (IoT) within the agricultural sector. Furthermore, the Internet of Things (IoT) plays a pivotal role in various application domains, including security, smart cities (e.g., smart traffic control systems), healthcare management, and more [1,2]. Through the utilisation of IoT sensors, a tremendous volume of data is collected, encompassing both unstructured as well as structured formats. The data obtained from sensors through the analysis of agricultural scenarios, such as images of fields, contributes to the resolution of numerous challenges in the agricultural sector[3]. Sophisticated data analytics is utilised within the agricultural sector to identify

[1]Research Scholar, Department of Computer Science and Engineering College of Engineering & Technology, Akola Akola, Maharashtra, India.
[2]Associate Professor, Department of Computer Science and Engineering, College of Engineering & Technology, Akola Akola, Maharashtra, India.

anomalies and detect diseases in pictures of crops or plants[4,5]. The integration of the Internet of Things (IoT) into the agricultural sector aims to improve the efficiency of sensor-based recognition methods. Precision farming, automated development of agriculture, and environmental forecasting are a few of the agricultural innovations that have emerged in recent years through the application of Artificial Intelligence (AI) as well as Machine Learning (ML) [6]. Modern cultivation utilising mechanised techniques increases the yield of higher-quality products, thereby assisting in the improvement of farmers' incomes [7]. This is particularly significant in India, where agriculture serves as the primary source of income for the majority of the population.

It is possible to determine the illness that affects plants by examining the leaves of the plants. This is due to the fact that the disorder of the plants is first impacted in the leaves. In the case of the plant that is completely afflicted, black spots appear on every leaf of the plant. In light of this, the identification of the illness by taking into account the leaf of the plant is very necessary in order to improve the yield of the produce. In this day and age, thanks to the progression of technology, automated devices are being produced for the detection of illness at a lesser cost. These gadgets are beneficial to farmers since they help them in increasing their productivity. The use of automated disease identification helps with the correct diagnosis and the administration of the most suitable pesticides to plants, which in turn helps to keep the environment from being polluted. In addition, the treatment of the plants against illness helps to safeguard the lives of the animals and insects that are dependent on the plants[8]. In addition, the cost that is used to carry out the therapy of the plant is decreased as a result of the correct treatment of the plants, which involves a more precise identification of the disease [9]. Multiple researchers developed a number of methods for the identification of plant diseases. These methods were based on the process of segmentation feature extraction, and classification, and they were accomplished via the use of image processing and soft computing[10-15].

When it comes to the old ways of detecting diseases in plant leaves, human intervention is crucial. It could be time-consuming and prone to error if the specialist in the field only looks at the plant to see whether it has any diseases. Because of this, the ability to identify diseases in plants' leaves instinctively is crucial for treating plants in a timely manner to increase production. Recently, soft computing algorithms for detection using identification and classification criteria were created with little time and cost. It makes use of optimisation techniques such as Genetic Algorithm (GA), Bacterial Foraging Optimisation (BFO), Fuzzy Logic (FL), Neural Network (NN), Particle Swarm Optimisation (PSO), and many more. This is where soft computing comes in handy, allowing for autonomous illness detection with little to no human interaction required [16]. To guarantee the model's convergence and consistency, a dynamic fuzzy system of control is created by merging neural networks employing fuzzy rules[17,18].

Therefore, the primary goal of this study is to use adaptive fuzzy control to create a reliable model for predicting plant diseases. The suggested fighter kitten optimisation is used to tweak the fuzzification and defuzzification layers' adjustable parameters, resulting in better judgements with less loss. The suggested prediction model's temporal complexity is minimised by processes like RoI extraction and feature extraction.

The remaining part of this proposed plant disease method is organised as follows: the assessment of the existing plant disease forecasting mechanism is discussed in Section 2, the system's model is described in section 3, and the suggested Cat swarm-based Harris Hawks (CSHH) optimisation method is detailed in Section 4. The examination of the suggested Cat swarm-based Harris Hawks (CSHH) Adaptive Fuzzy approach is outlined in Section 4, and Section 5 provides a summary of the findings.

2.Literature Review

The authors of [1] described a method for predicting plant leaf diseases using deep learning. This method takes into account multi-channel input, namely the picture of the leaf and the soil. The soil and leaf pictures were obtained and their characteristics were linked using the Pearson's correlation coefficient. The technique achieved a greater average accuracy, but it did not include feature extraction, which might further minimise computational cost and improve performance speed. [2] introduced an improved deep learning approach that incorporates a depth-wise separable layer. This method also addresses the problem of colour segregation and accurately identifies many illnesses by using the Gamma technique for picture rectification. The method's inability to assess performance with greater data is seen as a negative. The performance of deep learning is improved by using the AlexNet, surpassing the effectiveness of other conventional approaches [3-5]. The approach used the Adam optimizer to train and assessed the loss function using cross-entropy to achieve the optimal output. Nevertheless, the approach was unsuccessful in enhancing the velocity of the categorization process via supplementary modules. The illness propagation and forecasting were used by [6], whereby picture segmentation was performed to extract the key features. When it comes to using AI with DCNNs, there are a lot of obstacles, such as creating a model that fits the dataset, gathering a large dataset to train the model, figuring out how many layers to use, and calculating the amount of neurons in each layer. Furthermore, it is not easy to figure out how many parameters to feed into CNN[7-11].The technique exhibited high diagnostic accuracy and provided therapy suggestions; nevertheless, its performance was somewhat slower.

The proposed a method for predicting leaf disease using machine learning and segmentation, which was further improved via the use of optimisation techniques[12-15]. The complexity reduction was achieved by using background removal and picture segmentation. The classification was conducted using a multi-kernel machine learning technique, taking into account the leaves of medicinal plants. This approach achieved high accuracy in the classification process. However, additional improvements are required for its application-based implementation. The development of deep learning using fuzzy logic, as proposed by [8], involves the use of thresholding techniques to forecast the extent of the condition. The segmentation was performed using a threshold criterion, and a fuzzy-based judgement was made to determine the severity of the illness[16,17]. The method's validation was not used to demonstrate its robustness. For improved results, [18] used the deep learning approach with segmentation-based area of interest (ROI) mining to forecast diseases. In this study, illumination-based augmentation was used to enhance the identification of the varied backdrop. However, the accuracy improvement may be compromised when the number of epochs decreases throughout the model training process. The authors in [19,27] proposed a machine learning approach that incorporates depth learning. This method includes segmentation and extraction of features stages for reliable multi-classification. However, the lack of optimisation consideration results in the loss of information during classifier training, which can potentially degrade performance.

3. PROBLEM STATEMENT

The issues pertaining to the envisaged model of predicting plant leaf diseases are as follows:

- Image processing as well as measurements were used in the method described in [20] to identify the severity of diseases of plant leaves. An accuracy of 96% in detecting the severity of plant leaf diseases was achieved using this approach [20]. The problem with this approach was that it was only usable by those with scientific backgrounds or extensive experience using image processing [21].

- Otsu clustering thresholding relies on minimising variation within each class by choosing a threshold value to divide the picture into two groups. The distributions themselves cannot be adjusted for obvious reasons, however the variance of the two portions of the distribution may be modified by selecting a threshold value. The combined spread must be minimised, hence choosing a threshold is critical [23].
- Since barometric pressure is steady, its effects are hard to assess. However, barometric pressure along with rainfall impacts favourable predictions. Thus, barometric pressure impacts must be studied in numerous areas[27].
- The majority of traditional classifiers work best with smaller datasets and use hand-crafted picture features to sort data into groups. They are not good for handling very large datasets [22,24,26].

4.PROPOSED METHODOLOGY

The primary objective of the project is to create and implement an adaptive fuzzy expert system, using IoT technology, for the purpose of predicting leaf diseases in agricultural areas. At first, the IoT nodes gather the picture data from the field of farming and transmit it into the sink node. In the agricultural field, the sink node will serve as data aggregator, while the resulting picture data collection will serve as the input database for processing in order to identify the specific kind of plant leaf disease. The picture data will undergo basic pre-processing to eliminate any artefacts that may be visible on the image. Furthermore, the Region of interest (RoI) shall be retrieved from the pre-processed picture. The pre-processed picture will undergo a feature extraction method to extract important local directional ternary pattern (LDTP), locally directional and extremal pattern (LDEP), and Median ternary pattern (MTP) features. The retrieved features will be concatenated to generate the feature vector, which will then be inputted into the adaptive optimum fuzzy expert system. The suggested Cat swarm-based Harris Hawks (CSHH) optimisation technique will be used to build the rules of the fuzzy expert system [25] in an optimum manner. The CSHH optimisation method is created by integrating the key features of the cat swarm optimisation (CSO) algorithm [28,30] along with the Harris Hawks optimisation (HHO) algorithm [29].

The primary goals of the suggested paradigm are

- The objective is to create a highly efficient and adaptable expert system that use fuzzy logic to accurately forecast plant leaf illnesses. This system will utilise data obtained via the Internet of Things (IoT) to effectively regulate and mitigate the impact of these diseases on the whole plant.
- The objective is to create an efficient hybrid optimisation method called Cat swarm-based Harris Hawks (CSHH) optimisation algorithm. This algorithm will be used to optimise the parameters of the adaptive fuzzy expert system.
- To effectively manage the dynamic fuzzy expert framework in order to improve the effectiveness of the system in predicting plant leaf diseases.
- The goal is to put the model into action and make it possible to compare different approaches; this will show how the suggested model for plant leaf disease fares better.

4.1.System Architecture

Specifically, the Internet of Things sensors are used for the purpose of acquiring information on leaf disease, as seen in Figure 1. IoT nodes, cluster heads (CH), along with base stations (BS) are the components that make up the Internet of Things environment that is used for the information collecting. Radio link communication allows for direct connection between the nodes in the Internet of Things environment and the relevant CH. The nodes are dispersed equally across the environment. The transmission of data is sent to the base station (BS) via the corresponding CHs, where all of the nodes are positioned in a fixed location. Every node in the network's hierarchy has an unique ID for the purpose of data transmission, and the transfer of data is routed to the BS. Let us assume that the power of the terminals at the beginning of the network is denoted by the symbol, and that the network does not make use of the recharging process. Therefore, a certain amount of energy was lost at the nodes of the network together with the free space for each and every information transfer. In the CH node, the energy loss is represented by the notation,

$$\mathrm{E_{loss}}(\mathrm{Data_{CH}}^{\mathrm{x}}) = \mathrm{E_{EE}} * \mathrm{DB_{size}} \tag{1}$$

The energy that is lost at the regular node is denoted by E_{loss}, the size of the data packet is denoted by DB_{size}, the data that is sent from the node itself to the CH is denoted by $Data_{CH}$, and the energy that is lost as a result of electronic energy is denoted by E_{EE}. Up until the point when the node is rendered inoperable and its energy level drops below zero, the transmission of data packets continues.

Figure 1: IoT-based information gathering regarding the plant leaf disease

4.2.Proposed Cat swarm-based Harris Hawks (CSHH) optimisation based Adaptive Fuzzy Expert System for plant leaf disease prediction

It is possible for farmers to improve crop yield by avoiding the spread of plant diseases and improving the quality of the soil via the use of leaf images, which may be used to determine the presence of plant diseases in advance. The purpose of this project is to develop an automated framework for the prediction of leaf diseases relying on the AFES via the use of the Internet of Things in agricultural settings. In the beginning, the leaf pictures that are obtained by the sensors that are installed on agricultural fields are pre-processed in order to remove any artefacts and retrieve the Region of Interests (RoI) from the image that has been captured.

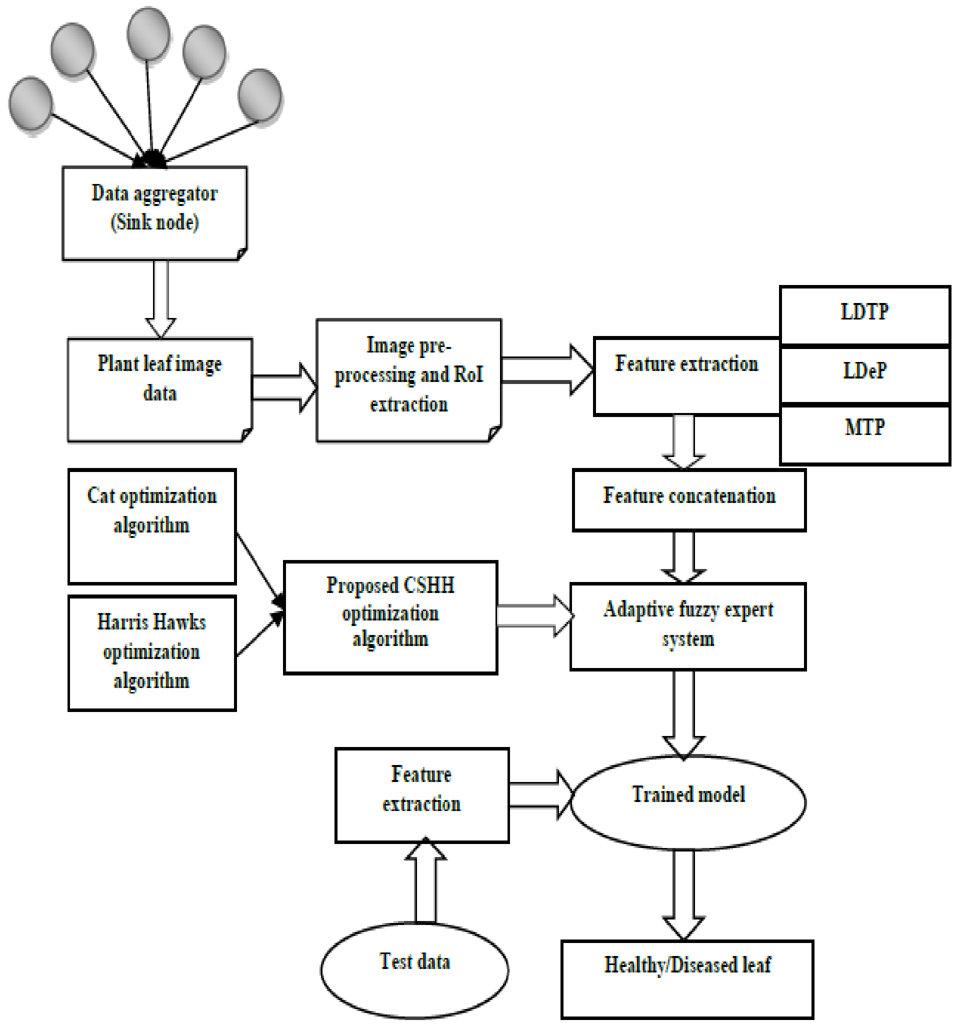


Figure 2. Proposed plant leaf disease prediction model

First, the relevant characteristics are retrieved by using the Median ternary pattern (MTP), followed by the local directional ternary pattern (LDTP), and last, the locally directed and external patterns are extracted. After the features have been retrieved, they are merged to produce the feature vector, which is then used by the adaptable fuzzy expert system to achieve the correct determination result. Furthermore, in order to acquire the accurate output, the rules of the fuzzy system for experts are ideally defined by means of the Cat swarm-based Harris Hawks (CSHH) optimisation method that has been suggested.

4.2.1. Data gathering

For the purpose of accurately monitoring the development of plants, the contemporary agricultural system necessitates the gathering of information about plants and the circumstances in which they are grown. The information that is provided by the data, which includes photos of plant leaves, relative humidity, moisture in the soil, and temperature, yields accurate information about the status of the plant's health and assists in the prediction of crop output. Therefore, the information that was described before is gathered straight from the field of agriculture by means of the sensors, and it will be kept in the nodes that are part of the

Internet of Things. Plant images data is extracted from the data that has been saved, and then it is subjected to further processing in order to conduct the plant leaf disease forecasting.

4.2.2. Image pre-processing and ROI extraction

The collected picture undergoes preprocessing and ROI extraction to enhance image quality and reduce model training time. The image preprocessing involves a range of procedures, including reshaping, colour correction, optimisation, and feature extraction, aimed at enhancing the quality of the picture. Extracting the Region of Interest (RoI) from the picture helps minimise the damage present in the image, hence enhancing the precision of the model. In addition, the RoI extraction procedure will reduce the computation time by removing unnecessary elements from the picture.

4.2.3.Feature Extraction

The important features are derived from the ROI to minimise computational overhead using the local directional ternary pattern (LDTP), Median ternary pattern (MTP), and the local directional and extremal pattern (LDEP) approach. A more complete explanation is provided below.

4.2.3.a.Local Directional Ternary Pattern

The local directional ternary pattern, also known as LDTP, is not affected by noise or illuminations and offers more useful characteristics with respect to edge reactions. In addition, the LDTP technique is more reliable in terms of the local primitives, and it takes into account nine edge responses. These edge responses correspond to the edge responses of the core pixels and the eight edge responses of the periphery pixels. A comparison is made between the core pixel and the periphery pixel at this stage of the encoding process in order to evaluate the directional pattern. In terms of mathematics, the LDTP may be expressed as follows:

$$TP_{LD} = \begin{cases} +1 \text{if} p_x \geq 0 \text{and} q_x \geq 0 \\ -1 \text{if} p_x \leq 0 \text{and} q_x \leq 0 \\ 0 \text{otherwise} \end{cases} \quad (2)$$

In this context, the centre pixel is denoted as p_x and q_x while the LDEP trait is denoted as TP_{LD}.

4.2.3.b.Median Ternary Pattern

Median filters are efficiently used for the extraction of features in the face of noise, resulting in precise information. A three-valued code is generated by taking into account the user-defined threshold, which guarantees strong performance against noise in regions that have a nearly uniform distribution. The assessment of the median pixel's intensity is based on the calculation of the intensity of nine pixels. Next, the pattern is used to establish three distinct values, and the expression used to generate the MTP code is known as,

$$PL_{F2}(r) = \begin{cases} +1 r > Med_k + Th \\ 0 Med_k - Th \leq r \leq Med_k + Th \\ -1 r < Med_k - Th \end{cases} \quad (3)$$

where, the local median is notated as Med_k, the threshold is notated as Th, the neighbor gray level is notated as r, and the MTP feature is notated as PL_{F2}.

4.2.3.c.Locally Directional and Extremal Pattern (LDEP)

The LDEP consists of two components: the Neighbor's Extremum-Related Local Pattern (NERLP) as well as the Directional Local Differential Count Patterns (DLDCP). The DLDCP-based information extraction method extracts both the magnitude as well as symbol information from the location of the centre pixel. The formula for obtaining the magnitude along with symbol information at the odd location is given as

$$\mathrm{OddPosition}_{SI} = \sum_{s=1}^{s\in odd} h\left(r_s(p,q) - r_k(p,q)\right) \quad (4)$$

$$\text{where, } r(a) = \begin{cases} 1 & a \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

where, the circumstance pixel gray value is notated as $r_s(p,q)$, and the gray value of the central pixel is notated as $r_k(p,q)$. Here, the symbol information of the odd position is referred to as $\mathrm{OddPosition}_{SI}$.

The magnitude of the odd position is notated as,

$$\mathrm{OddPosition}_{Mag} = \sum_{s=1}^{s\in odd} h\left(mag_s(p,q) - mean_k(p,q)\right) \quad (6)$$

$$\text{where, } mag_s(p,q) = |r_s(p,q) - r_k(p,q)| \quad (7)$$

Here, the difference between the magnitude of the $r_s(p,q)$ and $r_k(p,q)$ is referred to as $mag_s(p,q)$, the magnitude of odd position is notated as $\mathrm{OddPosition}_{Mag}$, mean of all median value in the input image is notated as $mean_k$.

Subsequently, the data pertaining to the magnitude and symbol associated with the even location is denoted as:

$$\mathrm{EvenPosition}_{SI} = \sum_{s=1}^{s\in even} h\left(r_s(p,q) - r_k(p,q)\right) \quad (8)$$

$$\mathrm{EvenPosition}_{Mag} = \sum_{s=1}^{s\in even} h\left(mag_s(p,q) - mean_k(p,q)\right) \quad (9)$$

where, the magnitude of the even position is notated as $\mathrm{EvenPosition}_{Mag}$ and the symbol information of the even position is notated as $\mathrm{EvenPosition}_{SI}$.

Following the extraction of the details for the both even and odd locations of the inputs, the NERLP is retrieved by taking into consideration the location along with difference patterns for the collection of information linked to the intensity. In this case, the extraction of the maximum intensity as well as location value is accomplished by utilising the phrase that follows:

$$\text{value1} = \max_{0\leq s\leq N-1}\left(r_s(p,q)\right) \quad (10)$$

$$\text{location1} = \arg\max_{0\leq s\leq N-1}\left(r_s(p,q)\right) \quad (11)$$

where, N refers to the circumjacent pixel. Then, the minimum value and the location are obtained as,

$$\text{value2} = \min_{0\leq s\leq N-1}\left(r_s(p,q)\right) \quad (12)$$

$$\text{location2} = \arg\min_{0\leq s\leq N-1}\left(r_s(p,q)\right) \quad (13)$$

Finally, the extreme difference pattern based on the symbol information and the magnitude is expressed as,

$$\text{Extremedifference}_{SI} = \begin{cases} \sum_{s=0}^{N-1} h\left(r_s(p,q)\right) - \left[\frac{\text{value1}+\text{value2}}{2}\right], W_h \le 2 \\ N + 1\text{otherwise} \end{cases} \quad (14)$$

where, $W_h$refers to the uniform pattern and the extreme difference pattern corresponding to the symbol information is notated as$\text{Extremedifference}_{SI}$.

$$\text{Extremedifference}_{mag} = \begin{cases} \sum_{s=0}^{N-1} h\left(i_s(p,q)\right) - i_k(j), W_i \le 2 \\ N + 1\text{otherwise} \end{cases} \quad (15)$$

where, $W_i$refers to the count of transition of numbers from 0 to 1 or 1 to 0, $i_k(j)$ refers to the mean for the input image, and the magnitude of the extreme difference is notated as$\text{Extremedifference}_{mag}$.

Thus, based on the NERLP and DLDCP features the LDEP features are obtained and are notated asPL_{F3}

4.2.4. Concatenation of Extracted Features

The characteristics that were collected from the earlier methodologies are concatenated in order to form a feature vector. This feature vector is then supplied to the AFES in order to categorise the plant leaf diseases. It is expressed as,

$$PL_F = \{PL_{F1}, PL_{F2}, PL_{F3}\} \quad (16)$$

where, the concatenated feature is denoted as PL_F, the MTP feature is notated as PL_{F2}, the LDEP feature is notated as PL_{F3} , and the LTEP feature is notated as PL_{F1} respectively.

4.3. Disease classification using Cat swarm-based Harris Hawks (CSHH) optimisation - based Adaptive Fuzzy Expert system

In order to accomplish the illness classification, the Adaptive Fuzzy Expert Systems (AFES) is used. Within this system, the changeable parameters that are utilised in the fuzzification as well as the de-fuzzification layers are modified by using the Cat swarm-based Harris Hawks (CSHH) optimisation technique that has been suggested.

4.3.1.Architecture of Adaptive Fuzzy Expert System

The illness prediction is carried out with the assistance of the Adaptive Fuzzy Expert Systems (AFES), which is seen in Figure 3. Not only does fuzzy-based illness prediction not need the involvement of an expert, but it also does not require more information. For this particular instance, the if-then rule is used in the suggested disease of plant leaves prediction in order to make the forecast of the illness. To improve the accuracy of the forecasts, the AFES utilises both the fuzzy rule and artificial intelligence (AI) in conjunction with one another. Additionally, the mistake reasons that are caused by the memorising of the data are minimised in the AFES, and it is more visible. Additionally, the learning process of the AFES in conjunction with optimisation helps to adjust the changeable factors in the first and fourth layers, which ultimately results in judgements that are more accurate while incurring the least amount of loss. A representation of the layout of the AFES may be seen in Figure 3.

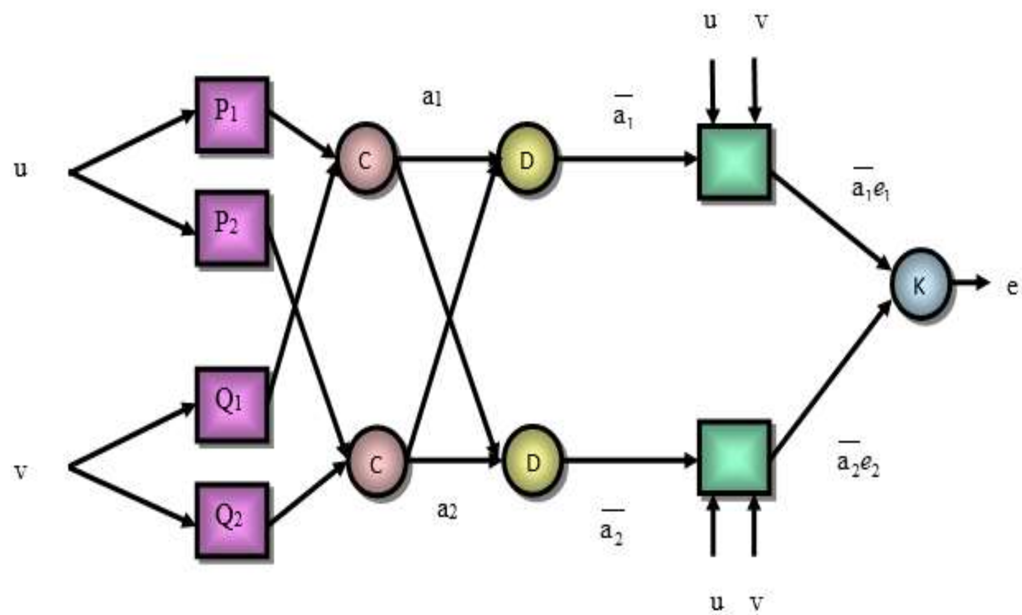


Figure 4: Architecture of Adaptive Fuzzy Expert System

The input that is obtained by the AFES is denoted by that, and the output that corresponds to that input is denoted by that. In this case, the fuzzy sets that were used for the purpose of building the parameters for the forecasting of the disease of plant leaves are denoted as accordingly. Furthermore, the outputs that were generated inside the fuzzy area are denoted as in accordance with the fuzzy rule that was determined by the suggested Cat swarm-based Harris Hawks (CSHH) optimisation method. Through the process of learning the AFES model, the parameters are optimised by using the Cat swarm-based Harris Hawks (CSHH) optimisation method that has been provided. Squares are used to represent both the adaptive nodes and the fixed nodes in Figure 4. The squares are used to represent the adaptive nodes.

4.3.1.a.Fuzzification:

According to the AFES, the fuzzification layer is composed of adaptive nodes. These nodes are responsible for providing the output by taking into account the degree of membership for the associated input. The expression for this layer is as follows:

$$A_t^1 = \alpha_{P_t}(u), t = 1,2 \tag{17}$$

$$A_t^1 = \alpha_{Q_{t-2}}(v), t = 3,4 \tag{18}$$

where, the linguistic label is notated as P_t, in which $\alpha_{P_t}(u)$, and $\alpha_{Q_{t-2}}(u)$ adopt the membership function, in which the values are obtained through the bell-shaped curve, in which "0" states to the minimum value and "1' denotes to the maximal value and is expressed as,

$$\alpha_{P_t}(u) = \frac{1}{1+\left\{\left(\frac{u-l_t}{n_t}\right)^2\right\}^{m_t}} \tag{19}$$

where, the premise parameters are notated as n_t, $m_t$and$l_t$respectively.

4.3.1.b.Multiplier$(\mathbf{C})$: This layer makes use of simple multiplication, which results in the product being obtained via the use of fixed nodes. This layer is responsible for producing the product that is expressed as

$$A_t^2 = a_t = \alpha_{P_t}(u)\alpha_{Q_t}(v)t = 1,2 \tag{20}$$

here, the term $a_t$refers to the fire strength of the rule.

4.3.1.c.Normalization$(\mathbf{D})$: This layer makes use of the aggregation of the fire intensity rule by calculating the ratios of fire intensity that correlate to the node. The expression for this rule is as follows:

$$A_t^3 = \overline{a_t} = \frac{a_t}{a_1 + a_2} t = 1,2 \tag{21}$$

Thus, the output acquired from this layer represents the normalized fire strengths.

4.3.1.d.Defuzzification: Using the adaptive nodes that are used in this layer, the normalised firing strengths are multiplied by the first-order polynomial, and the results are represented as follows:

$$A_t^4 = \overline{a_t}e_t = a_t(x_t u + y_t v + z_t)t = 1,2 \tag{22}$$

here, x_t, y_t, and$z_t$refers to the consequent parameters.

4.3.1.e.Output: For the purpose of predicting the plant leaf disease, the calculation of the total of the overall outputs is carried out and is stated as

$$A_t^5 = \sum_{t=1}^{2} \overline{a_t}e_t = \frac{\sum_{t=1}^{2} a_t e_t}{a_1 + a_2} \tag{23}$$

Here, the tunable parameters utilized in the fuzzification and the defuzzification layers like $n_t, m_t$and$l_t$and $x_t, y_t$and$z_t$are tuned using the Cat swarm-based Harris Hawks (CSHH) optimisation algorithm for enhancing the performance of the AFES model.

4.3. Proposed Cat swarm-based Harris Hawks (CSHH) optimisation algorithm

For the purpose of developing the Cat swarm-based Harris Hawks (CSHH) optimisation algorithm, the food-catching behaviour of the Hawks [16] and the awareness of the Cats [17] in migrating from one position to another area in quest of food are combined and incorporated.

The suggested Hawks Cat optimisation method considers the cooperative attacking technique of hawks and their various pursuit patterns. Hawks wait on higher perches or trees in the early twilight to find their prey. Hawks may use a leapfrog movement to capture the prey by flying over the targeted region using split as well as rejoin criteria. Hawks can use coordinated assault to catch the target from all angles, limiting the quarry's escape. Thus, Hawks use numerous hunting methods to obtain prey. Hawks seek the specified area for the quarry and then try to catch it using different assault methods. Hawks' hunting approach is improved by ignoring prey capture awareness. To improve predation, the cat's awareness likelihood is combined with the Hawks' food-capturing behaviour.

Cats may be awake even while napping by opening their eyelids. Cats move slowly and cautiously or remain in the same place with heightened attention to avoid assaults. In the Hawks Cat optimisation system, the cat's awareness is combined with the Hawks' food-capturing behaviour to prevent prey escape. Thus, cautious prey acquisition with high vigilance is quicker and less likely to escape, ensuring rapid convergence and improved exploitation. Training the

AFES model to identify illness is difficult owing to significant learning loss. By fine-tuning the premises and variables at the fuzzification along with de-fuzzification phases of the AFES model, Hawks Cat optimisation minimises loss in prediction model learning. The suggested Hawks Cat optimisation improves plant forecast accuracy by balancing intensification and diversity requirements with cat and Hawk awareness.

4.3.1.Steps of Proposed Cat swarm-based Harris Hawks (CSHH) optimisation Algorithm

This section provides a full explanation of the processes used to discover the best tuning solution for the AFES in plant disease forecasting using leaf analysis.

Step 1: Initialization of Parameters : Initializing the Hawk's population in the feature space, the total population size is denoted as T_P, and the maximum iteration is specified as Max_{iter}.

Step 2: Fitness Function: The fitness is computed for every hawk in the fitness set in order to provide an updated location. It serves as a function to address optimisation problems and is denoted as follows:

$$\mathrm{Fit\_Func} = \frac{\mathrm{FF_{tp}} + \mathrm{FF_{tn}}}{\mathrm{FF_{tp}} + \mathrm{FF_{tn}} + \mathrm{FF_{fp}} + \mathrm{FF_{fn}}} \tag{24}$$

whereby the fitness is expressed as Fit_Func, the true positive as $\mathrm{FF_{tp}}$, the true negative as $\mathrm{FF_{tn}}$, the false positive as $\mathrm{FF_{fp}}$, and false negatives as $\mathrm{FF_{fn}}$.

Algorithm 1: Pseudo-code for proposed Cat Swarm-Based Harris Hawks (CSHH)Optimization

Pseudo-code for proposed Cat Swarm-Based Harris Hawks (CSHH)Optimizatio
$\mathrm{H(i)}$, N and $\mathrm{iter_{max}}$
$\mathrm{H(i+1)}$
Initialize the population with population size N
While $\mathrm{i \le iter_{max}}$
Estimate the Hawk's fitness value
Set $\mathrm{H_{prey}(i)}$ as the best location of the prey
Update the initial energy $\mathrm{EG_I}$and jump strength M
Update Zas per equation (27)
if $(|\mathrm{Z}| \ge 1)$then
Diversification phase
Update the position using (25)
if $(|\mathrm{Z}| < 1)$
Intensification phase
If $\mathbf{w} \ge \mathbf{0.5}$and $\mathbf{Z} \ge \mathbf{0.5}$then
Update the position using (28)
If $\mathbf{w} \ge \mathbf{0.5}$and $\mathbf{Z} < \mathbf{0.5}$then
Update the position using (31)
else If $\mathbf{w} \ge \mathbf{0.5}$and $\mathbf{Z} \ge \mathbf{0.5}$then
Update the position using (32)
else If $\mathbf{w} < \mathbf{0.5}$and $\mathbf{Z} < \mathbf{0.5}$then
Update the position using (33)
Integrate the characteristics of a kitten

If $\mathbf{w} \geq \mathbf{0.5}$and $\mathbf{Z} \geq \mathbf{0.5}$then
Update the position using (40)
If $\mathbf{w} \geq \mathbf{0.5}$and $\mathbf{Z} < \mathbf{0.5}$then
Update the position using (42)
else If $\mathbf{w} \geq \mathbf{0.5}$and $\mathbf{Z} \geq \mathbf{0.5}$then
Update the position using (43)
else If $\mathbf{w} < \mathbf{0.5}$and $\mathbf{Z} < \mathbf{0.5}$then
Update the position using (44)
Return $\mathrm{H}(\mathrm{i}+1)$

Step 3: Differentiation: During this stage, sitting haphazardly on towering trees or poles serves as the quarry hunt. Given the possibility of locating the prey be P, the hawk's location during the differentiation phase is stated as,

$$\mathrm{H}(\mathrm{i}+1) = \begin{cases} \mathrm{H_d}(\mathrm{i}) - \mathrm{a_1}|\mathrm{H_d}(\mathrm{i}) - 2\mathrm{a_2}\mathrm{H}(\mathrm{i})| \mathrm{b} \geq 0.5 \\ [\mathrm{H_{prey}}(\mathrm{i}) - \mathrm{H_c}(\mathrm{i})] - \mathrm{a_3}[\mathrm{Y} + \mathrm{a_4}(\mathrm{X} - \mathrm{Y})] \mathrm{b} < 0.5 \end{cases} \tag{25}$$

When the hawk's position is indicated as $\mathrm{H}(\mathrm{i}+1)$, the quarry's location is indicated as $\mathrm{H_{prey}}(\mathrm{i})$,the parameters' a_1,a_2,a_3,$a_4$random number range is [0,1], and the associated iteration is indicated as b. The hawk's average location is shown as $\mathrm{H_c}(\mathrm{i})$ while the hawk that was selected at random is indicated as $\mathrm{H_d}(\mathrm{i})$. The top and lower borders of the feature space are referred to as X and Y are boundaries.

Step 4: Phase changeover: The energy of the prey determines when the hawk phase shifts from diversification to intensification, and the formula for calculating the energy of the prey is as follows:

$$\mathrm{EG_I} = 2\mathrm{EG_I}\left(1 - \frac{\mathrm{iter}}{\mathrm{iter_{max}}}()\right) \tag{27}$$

where EG_I is the prey's starting energy, $iter_{max}$ denotes the maximum iteration, and EG represents the prey's energy needed to escape the hawk.

Step 5: Intensification: In order to catch the prey without escaping, the group of hawks attacks them concurrently from several angles in an attempt to grab the target. The hawk's propensity to escape the quarry is indicated by K. The likelihood of fleeing away from a hawk is higher when the K value is larger than 0.5. Otherwise, the likelihood of escape is lower. Here, four distinct tactics are taken into consideration for the amplification of the prey in the suggested Cat swarm-based Harris Hawks (CSHH) optimization.

5.Experimental setup

This section provides detailed information on the software requirements, dataset, and performance measures used in the study of the illness prediction model. The illness detection model is implemented using MATLAB, specifically the MATLAB 2020a programme running on a Windows 10 operating system with 8 GB of memory. The newly acquired plant disease collection comprises 87,000 photos depicting both infected and healthy crops. The whole dataset is categorised into 38 distinct groups.The study employs measurements such as specificity, accuracy, and sensitivity to evaluate performance.

The following approaches are utilised: Support Vector Machines (SVM), Artificial Neural Networks (ANN), AlexNet with transfer learning, K-Nearest Neighbours (KNN), Adaptive Neuro-Fuzzy Inference System (ANFIS), Multi-column Convolutional Neural Network (MCNN), Genetic Algorithm-tuned Adaptive Feature Extraction System (GA-tuned AFES), Cuckoo Search Optimization-tuned Adaptive Feature Extraction System (CSO-tuned AFES), and Harris Hawks Optimization-tuned Adaptive Feature Extraction System (HHO-tuned AFES).

5.1.Experimental analysis

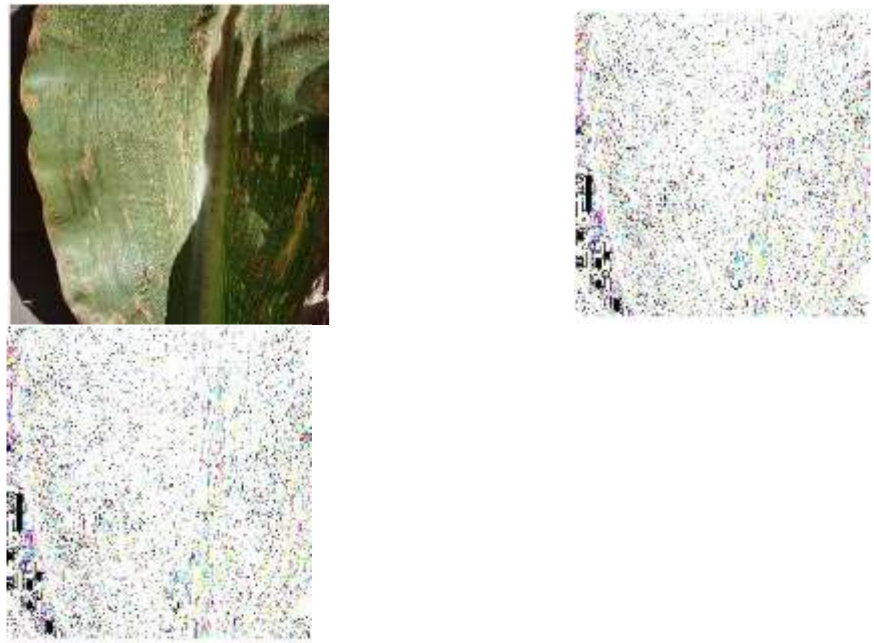
The outcomes obtained from samples of the the grape data set, the tomato data set, and Corn datasets are shown in Figure 4, 5, and 6 correspondingly. The LDTP and LDEP techniques are used to extract the appropriate characteristics from the input signals. These features are then inputted into the FKO-AFES model to accurately determine the presence of disease in the plant.

(a) (b) (c)

Figure 5. Analysis of Grape dataset a) sample input b) LDTP output c) LDEP output

(a) **(b)** **(c)**

Figure 6. Analysis of tomato dataset a) sample input b) LDTP output c) LDEP output



(a) (b) (c)

Figure 7. Analysis using Corn dataset a) sample input b) LDTP output c) LDEP output

The comparison analysis is determined by altering the K-fold and training percentage. The dataset pertaining to Corn, Grape, along with Tomato is employed for comparative analysis.

6.Results and Discussion

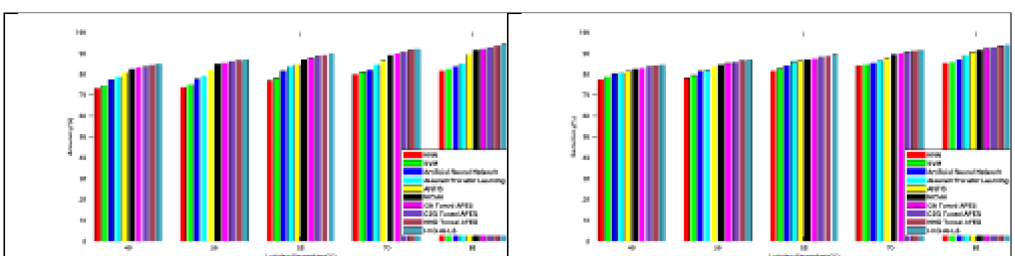
6.1.Using Grape dataset

6.1.a.Analysis of training percentages using Grape dataset

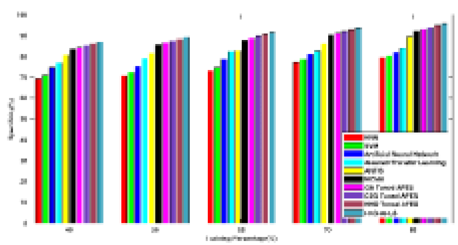
Figure 8 shows Grape dataset comparisons by training %. It is shown that the FKO-AFES model is more accurate than competent approaches. For instance, the KNN model obtains its highest efficiency at 80% of training, 81.6350%, whereas the suggested model achieves 94.4100%, a 13.5314% increase.

The standard KNN model has the best sensitivity at 80% of training 85.3393%, while the FKO-AFES-based disease detection model has the highest accuracy at 94.1094%. This reveals that FKO-AFES outperforms KNN by 9.3190%.

The FKO-AFES model's specificity peaks at 95.6548 % during 80% training. However, the typical KNN model has 79.3580% maximum specificity. This shows that the FKO-AFES model improves 17.0371%.



a) **b)**



c)

Figure 8. Training percentage analysis using grape dataset a) accuracy, b) sensitivity, and c) specificity

6.1.b. K-fold analysis for grape dataset

In figure 10 a), b), and c), the K-fold value is varied to compare the three parameters using a grape dataset. The graphic shows that the suggested model is more accurate than standard techniques. The KNN model has the greatest accuracy at K-fold 10, 83.613%, whereas the suggested model has 96.1793%, a 13.0648% performance gain.

The classic KNN model has the greatest sensitivity at K-fold 10 of 87.9602%, whereas the suggested FKO-AFES-based disease prediction model has the highest accuracy of 96.4859%. This reveals that the FKO-AFES model outperforms the KNN model by 8.8361%.

Specificity for the FKO-AFES model peaks at 96.8345% at K-fold 10. However, the standard KNN model has 81.2669% maximum specificity. The suggested FKO-AFES model improves performance by 16.0765%.

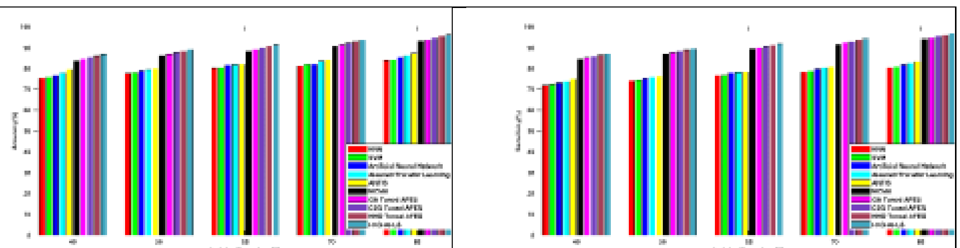
6.2.Using Tomato dataset

6.2.a.Training percentage analysis using tomato data

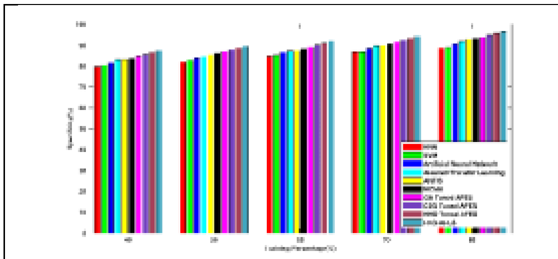
Figure 9 shows a tomato dataset comparison of training percentages. Figure shows that the suggested FKO-AFES model is more accurate than competent techniques. For instance, the KNN model has the greatest efficiency at 80% of training, 83.8350%, whereas the suggested model has 96.0302%, a 12.6993% performance gain.

Sensitivity is best for the classic KNN model at 80% of training 80.3416%, whereas accuracy is highest for the suggested FKO-AFES-based disease detection model at 96.5349%. This reveals that the FKO-AFES model outperforms the KNN model by 16.7745%.

Specificity for the FKO-AFES model peaks at 96.4859 % at 80% training. The standard KNN model has 88.8020% maximum specificity. This shows that the FKO-AFES model improves performance by 7.9637%.



a)	b)



c)

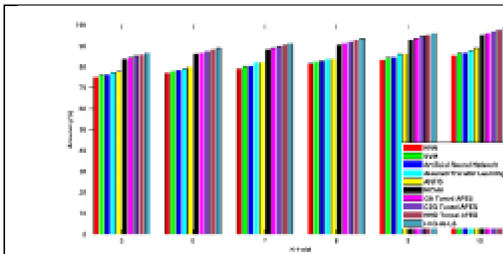
Figure 9. Training percentage analysis using tomato dataset a) accuracy, b) sensitivity, and c) specificity

6.2.b. K-fold analysis using the grape dataset

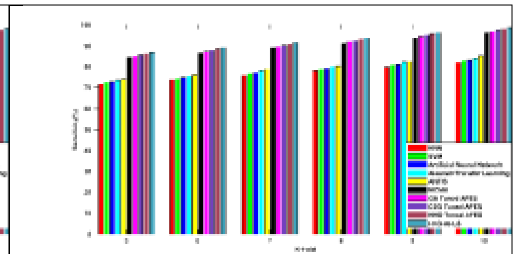
In figure 10 a), b), and c), the K-fold value is varied to compare the three parameters using a grape dataset. The graphic shows that the suggested model is more accurate than standard techniques. The KNN model has the greatest accuracy at K-fold 10, 85.3598%, while the suggested model has 98.1357%, a 13.0184% performance gain.

The classic KNN model has the greatest sensitivity at K-fold 10 of 87.9602%, whereas the suggested FKO-AFES-based disease prediction model has the highest accuracy of 96.4859%. This reveals that the FKO-AFES model outperforms the KNN model by 8.8361%.

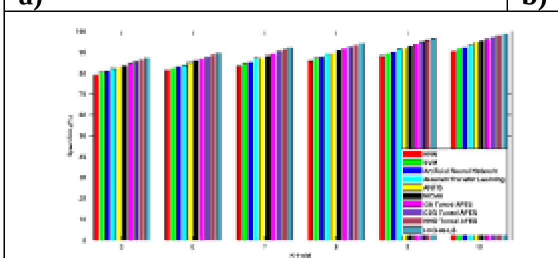
The FKO-AFES model has a maximum specificity of 81.9038% at K-fold 10. The standard KNN model has 98.4851% maximum specificity. This shows that the FKO-AFES model improves performance 16.8363%.



a)



b)



c)

Figure 10. K-fold analysis using tomato dataset a) accuracy, b) sensitivity, and c) specificity

6.3. Using corn dataset

6.3.a.Analysis of training percentages using Corn data

Figure 11 illustrates the comparative analysis of the maize dataset by varying the training %. The figure 11 a) displays a comparative comparison regarding correctness. The suggested model demonstrates a greater level of accuracy compared to standard approaches. As an example, the KNN model achieves its highest accuracy of 80% during training. The maximum accuracy of the KNN model is determined to be 70.4232%. On the other hand, the suggested model achieves a maximum accuracy of 94.6055%, indicating a performance improvement of 25.56%.

Regarding sensitivity, the conventional KNN model achieves its greatest value at 80% of training, namely 77.5518%. On the other hand, the suggested FKO-AFES-based plant disease model achieves a maximum accuracy of 92.9672%. This demonstrates that the suggested FKO-AFES model exhibits a performance enhancement of 16.5815% when compared to the KNN model.

The suggested FKO-AFES model achieves an exceptional specificity of 98.1945% for 80% of the training period. Nevertheless, the standard KNN model has a maximum specificity of 64.4596%. The suggested FKO-AFES model achieves a performance improvement of 34.3646%.

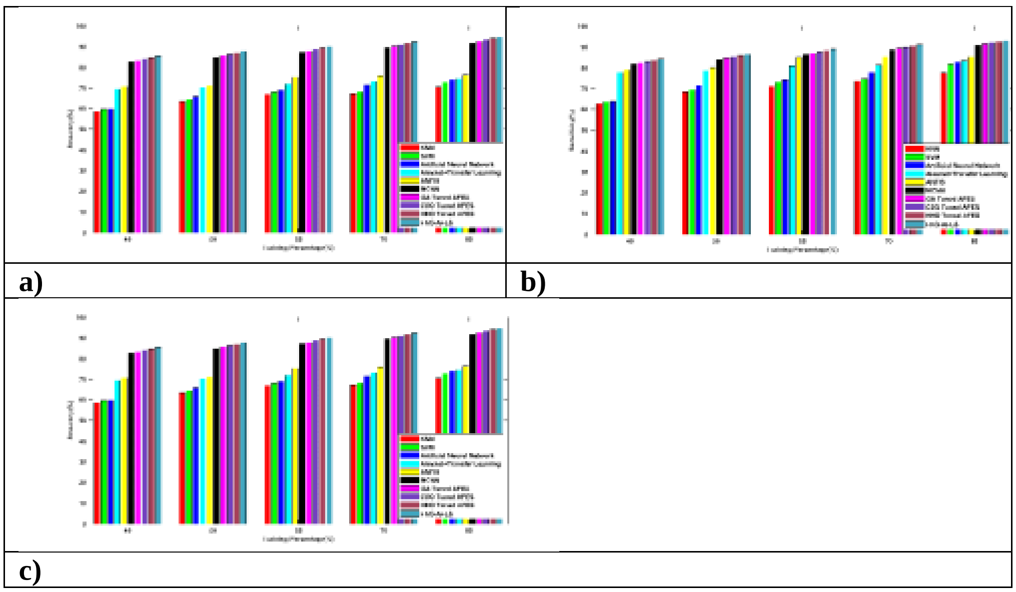


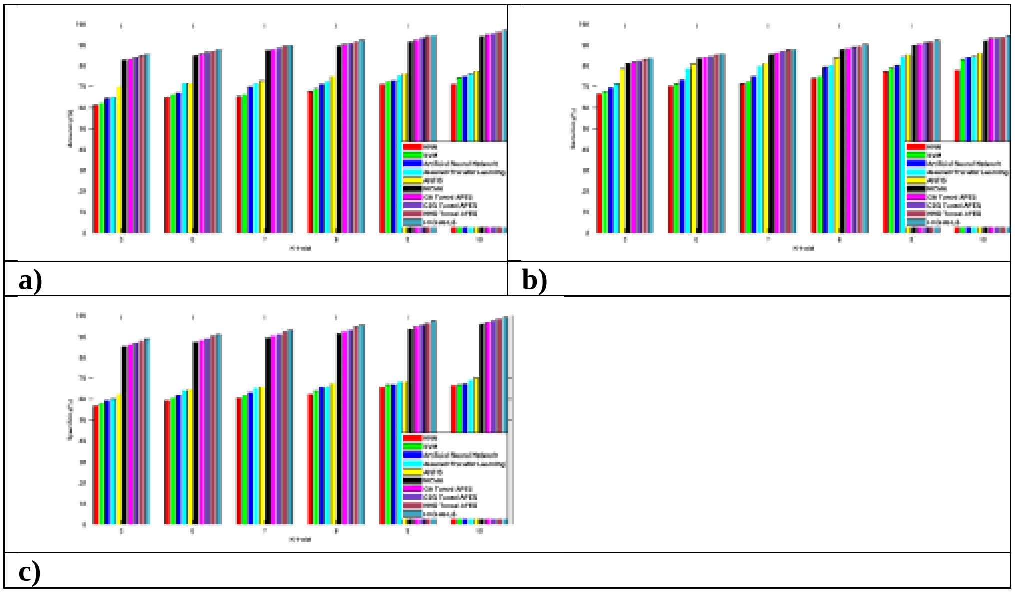
Figure 11. Training percentage analysis using corn dataset a) accuracy, b) sensitivity, and c) specificity

6.3.b. K-fold analysis with the corn dataset

Figure 12 a), b), and c) illustrate the comparative study of sensitivity, specificity, as well as accuracy by altering the K-fold value using a maize dataset. The suggested model demonstrates a greater level of accuracy compared to standard approaches. For example, the K-nearest neighbours (KNN) model has the best accuracy when using a 10-fold cross-validation, with a maximum accuracy of 71.1523%. In contrast, the suggested model achieves a maximum accuracy of 96.8661%, showing an improvement of 26.54%.

The classic KNN model achieves a sensitivity of 77.8017% at K-fold 10, whereas the suggested FKO-AFES-based plant illness model achieves a maximum accuracy of 94.3242%. This demonstrates that the suggested FKO-AFES model has a 17.5167% enhancement when compared to the KNN model.

The FKO-AFES model achieves a maximum specificity of 99.00% at K-fold 10. Nevertheless, the standard KNN model achieves a maximum specificity of 66.4596%. The FKO-AFES model achieves a 32.8690% improvement.



a) b) c)

Figure 12. K-fold analysis using corn dataset a) accuracy, b) sensitivity, and c) specificity

6.4.Comparative analysis:

This section provides a detailed analysis of the outcomes achieved via the methodologies used in the detection of plant leaf diseases.

Table 1. Comparative analysis of Existing Methods with the Proposed Approach for Corn Dataset

Methods	Accuracy		Sensitivity		Specificity	
	Training percentage (80%)	K-fold	Training percentage	K-fold	Training percentage	K-fold
KNN	70.4232	71.15232	77.55188	77.80171	66.45968	64.45031
SVM	72.7361	74.19295	81.72288	82.96141	66.73076	64.89039
ANN	73.9436	74.86482	82.89485	84.06052	67.06944	66.30558
AlexNet + transfer learning	74.5474	75.84902	83.48084	84.28767	68.95006	66.93278
ANFIS	76.3047	77.36416	85.24191	86.14841	70.10118	69.03717
MCNN	91.7329	94.01794	90.59865	92.08802	95.61328	94.54918
GA tuned AFES	92.4459	94.72729	91.47108	93.04214	96.58347	95.46854
CSO tuned AFES	93.1588	95.44024	92.08372	93.11615	97.49148	96.3879
HHO tuned AFES	93.8718	96.15687	92.4313	93.63191	98.33554	97.30726
FKO-AFES	94.6055	96.86614	92.96727	94.32425	99	98.19453

The techniques used for comparison include ANN, SVM, KNN, AlexNet with transfer learning, ANFIS, MCNN, AFES with GA tuning, AFES with CSO tuning, and AFES with HHO tuning. The assessment is conducted using three datasets: Corn, Grape and Tomato.

Table 2. Comparative analysis of Existing Methods with the Proposed Approach for Grape Dataset

Methods	**Accuracy**		**Sensitivity**		**Specificity**	
	Training percentage (80%)	**K-fold**	**Training percentage**	**K-fold**	**Training percentage**	**K-fold**
KNN	81.635	83.6137	85.33933	87.96022	79.35802	81.26693
SVM	82.38846	84.58899	85.81942	88.15476	80.32274	82.9877
ANN	83.72089	85.86874	87.01711	89.15571	81.96463	84.35217
AlexNet + transfer learning	84.94035	88.36284	89.09529	91.56205	83.94059	86.93188
ANFIS	89.5035	88.64398	90.60663	92.13953	89.29541	87.0181
MCNN	91.39878	93.05351	91.63784	94.01434	92.07372	93.02322
GA tuned AFES	92.15663	93.80014	92.25573	94.63223	92.9791	93.90606
CSO tuned AFES	92.77597	94.54678	92.87362	95.25012	93.60607	94.7889
HHO tuned AFES	93.65727	95.29341	93.49151	95.86801	94.75959	95.67174
FKO-AFES	94.41009	96.17935	94.1094	96.4859	95.65488	96.83458

The results are shown in Tables 1, 2, and 3, respectively. To conduct training percentage analysis, the training percentage is adjusted within the range of 40% to 80%. The value of K in K-fold analysis is changed within the range of 5 to 10. The research demonstrates that the suggested model has the maximum performance in terms of sensitivity, specificity, as well as accuracy when trained with 80% of the data and using a value of k=5.

Table 2. Comparative analysis of Existing Methods with the Proposed Approach for Tomato Dataset

Methods	**Accuracy**		**Sensitivity**		**Specificity**	
	Traing percentage (80%)	**K-fold**	**Training percentage**	**K-fold**	**Training percentage**	**K-fold**
KNN	83.83501	85.35987	80.34158	81.90386	88.80204	90.287
SVM	84.02561	86.37921	80.60642	82.78749	88.82231	91.55891
ANN	85.38452	86.70553	81.73136	83.01704	90.63794	92.08611

AlexNet + transfer learning	86.26011	87.57947	82.48023	83.90508	91.75555	93.05973
ANFIS	87.23219	88.83866	83.18473	85.08353	92.88552	94.5221
MCNN	93.08914	95.0833	93.99478	95.96238	93.1144	95.15506
GA tuned AFES	93.73639	95.84637	94.63346	96.59307	93.77669	96.05813
CSO tuned AFES	94.50101	96.60943	95.26727	97.22376	94.67976	96.9612
HHO tuned AFES	95.26563	97.3725	95.90109	97.85446	95.58283	97.86427
FKO-AFES	96.03025	98.13557	96.5349	98.48515	96.4859	98.76734

Based on the discussion, it is evident that the suggested model achieves superior performance as a result of the effective development of the ANFES classifier using the FKO method.

CONCLUSION

Forecasting plant diseases is an essential undertaking in precision agriculture as it facilitates the mitigation of the dissemination of plant leaf diseases. An adaptable fuzzy expert system is used in the development and analysis of a plant disease forecasting technique that is enabled by the Internet of Things (IoT). The illness prediction model based on AFES assures optimal performance by integrating the features acquired by approaches such as LDTP, MTP, and LDEP. In addition, the adjustable parameters used in the fuzzification and de-fuzzification layers are calibrated using the suggested Fighter Kitten optimisation technique, hence enhancing the precision of the outcome while minimising the loss function. The experimental findings indicate that the proposed FKO-AFES achieves a performance increase of 26.54%, 17.5617%, and 32.8690% compared to the current KNN model when applied to the maize dataset. However, the pattern recognition ability of the AFES used in the study is somewhat lower compared to that of the neural network. Moreover, the training of the AFES model is a laborious procedure that amplifies the temporal complexity. Therefore, in order to address these problems in the future, ensemble learning techniques that can accurately identify patterns and require little training time will be used to improve the performance of predictive models.

REFERENCES

1. Dhandapani, P. and Varadarajan, A., "Multi-Channel Convolutional Neural Network for Prediction of Leaf Disease and Soil Properties", International Journal of Intelligent Engineering and Systems, vol.15, no.1, pp.318-328, 2022.
2. Lilhore, U.K., Imoize, A.L., Lee, C.C., Simaiya, S., Pani, S.K., Goyal, N., Kumar, A. and Li, C.T., "Enhanced convolutional neural network model for cassava leaf disease identification and classification", Mathematics, VOL.10, NO.4, p.580, 2022.
3. Zou, Xiuguo, Wenchao Liu, Zhiqiang Huo, Sunyuan Wang, Zhilong Chen, Chengrui Xin, Yungang Bai et al. "Current Status and Prospects of Research on Sensor Fault Diagnosis of Agricultural Internet of Things." Sensors 23, no. 5 (2023): 2528.
4. Thakur, Poornima Singh, Pritee Khanna, Tanuja Sheorey, and Aparajita Ojha. "Trends in vision-based machine learning techniques for plant disease identification: A systematic review." Expert Systems with Applications (2022): 118117.
5. Chen, H.C., Widodo, A.M., Wisnujati, A., Rahaman, M., Lin, J.C.W., Chen, L. and Weng, C.E., "AlexNet convolutional neural network for disease detection and classification of tomato leaf". Electronics, vol.11, no.6, pp.951, 2022.

6. Alqethami, S., Almtanni, B., Alzhrani, W. and Alghamdi, M., "Disease Detection in Apple Leaves Using Image Processing Techniques", Engineering, Technology & Applied Science Research, vol.12, no.2, pp.8335-8341, 2022.
7. Sumithra, M.G. and Saranya, N., "Particle Swarm Optimization (PSO) with fuzzy c means (PSO-FCM)–based segmentation and machine learning classifier for leaf diseases prediction", Concurrency and Computation: Practice and Experience, vol.33, no.3, pp.e5312, 2021.
8. Sibiya, Malusi, and Mbuyu Sumbwanyambe, "Automatic Fuzzy Logic-Based Maize Common Rust Disease Severity Predictions with Thresholding and Deep Learning", Pathogens 10, no. 2, pp. 131, 2021.
9. Hossain, Syed Mohammad Minhaz, Kaushik Deb, Pranab Kumar Dhar, and Takeshi Koshiba, "Plant Leaf Disease Recognition Using Depth-Wise Separable Convolution-Based Models", Symmetry, vol. 13, no. 3, pp. 511, 2021.
10. Nandhini, S. and Ashokkumar, K., "Machine Learning Technique for Crop Disease Prediction Through Crop Leaf Image", Applied Mathematics, vol.16, no.2, pp.149-158.
11. Garg, Disha, and Mansaf Alam, "Deep Learning and IoT for Agricultural Applications", In Internet of Things (IoT), pp. 273-284, 2020.
12. Reddy, G. Rama Mohan, Nettam Sai Sumanth, and N. Sai Preetham Kumar, "plant leaf disease detection using cnn and raspberry pi", International Journal, vol. 5, no. 2, 2020.
13. Baranwal, Tanmay, and Pushpendra Kumar Pateriya, "Development of IoT based smart security and monitoring devices for agriculture", In proceedings of 2016 6th International Conference-Cloud System and Big Data Engineering (Confluence), IEEE, pp. 597-602, 2016.
14. Mahakud, Rina, Binod Kumar Pattanayak, and Bibudhendu Pati. "Challenges and Solution for Identification of Plant Disease Using IoT and Machine Learning." Specialusis Ugdymas 1, no. 43 (2022): 4789-4810.
15. Mishra, Monalisa, Prasenjit Choudhury, and Bibudhendu Pati. "IoT enabled plant leaf disease segmentation and multi-classification using mayfly bald eagle optimization-enabled machine learning." Multimedia Tools and Applications (2024): 1-35.
16. Islam, Shahidul, Sanjay Jamwal, Mahmood Hussain Mir, and Qamar Rayees Khan. "IoT-Smart Agriculture: Comparative Study on Farming Applications and Disease Prediction of Apple Crop using Machine Learning." Iraqi Journal of Science (2022): 5520-5533.
17. Kumar, Rajesh, Deepak Sinwar, Amit Pandey, Tesfaye Tadele, Vijander Singh, and Ghanshyam Raghuwanshi. "IoT Enabled Technologies in Smart Farming and Challenges for Adoption." Internet of Things and Analytics for Agriculture, Volume 3 (2022): 141-164.
18. Orchi, Houda, Mohamed Sadik, and Mohammed Khaldoun. "On using artificial intelligence and the internet of things for crop disease detection: A contemporary survey." Agriculture 12, no. 1 (2021): 9.
19. Kussul, Nataliia, Mykola Lavreniuk, Sergii Skakun, and Andrii Shelestov, "Deep learning classification of land cover and crop types using remote sensing data", IEEE Geoscience and Remote Sensing Letters, vol. 14, no. 5, pp. 778-782, 2017.
20. Chouhan, Siddharth Singh, Ajay Kaul, Uday Pratap Singh, and Sanjeev Jain, "Bacterial foraging optimization based radial basis function neural network (BRBFNN) for identification and classification of plant leaf diseases: An automatic approach towards plant pathology", IEEE Access, vol. 6, pp. 8852-8863, 2018.
21. Sakhamuri, Sridevi, and Vijaya Sri Kompalli. "An overview on prediction of plant leaves disease using image processing techniques." In IOP Conference Series. Materials Science and Engineering, vol. 981, no. 2. IOP Publishing, 2020.
22. Dhaka, Vijaypal Singh, Sangeeta Vaibhav Meena, Geeta Rani, Deepak Sinwar, Muhammad Fazal Ijaz, and Marcin Woźniak. "A survey of deep convolutional neural networks applied for prediction of plant leaf diseases." Sensors 21, no. 14 (2021): 4749.
23. Khan, Tahir. "AN AGRICULTURAL INTERNET OF THINGS (A-IOT) BASED INTELLIGENT SYSTEM FOR DISEASE PREDICTION USING TRANSFER LEARNING, A CASE STUDY." Lahore Garrison University Research Journal of Computer Science and Information Technology 7, no. 3 (2023).

24. Chouhan, Siddharth Singh, Uday Pratap Singh, and Sanjeev Jain. "Automated plant leaf disease detection and classification using fuzzy based function network." Wireless Personal Communications 121 (2021): 1757-1779.
25. Barbedo, Jayme Garcia Arnal, "An automatic method to detect and measure leaf disease symptoms using digital image processing", Plant Disease, vol. 98, no. 12, pp. 1709-1716, 2014.
26. Gadekallu, Thippa Reddy, Dharmendra Singh Rajput, M. Praveen Kumar Reddy, Kuruva Lakshmanna, Sweta Bhattacharya, Saurabh Singh, Alireza Jolfaei, and Mamoun Alazab, "A novel PCA–whale optimization-based deep neural network model for classification of tomato plant diseases using GPU", Journal of Real-Time Image Processing, pp. 1-14, 2020.
27. Ahmed, Aram M., Tarik A. Rashid, and Soran Ab M. Saeed. "Cat swarm optimization algorithm: a survey and performance evaluation." Computational intelligence and neuroscience 2020 (2020).
28. Seyyedabbasi, Amir, and Farzad Kiani. "Sand Cat swarm optimization: A nature-inspired algorithm to solve global optimization problems." Engineering with Computers 39, no. 4 (2023): 2627-2651.
29. Heidari, A.A., Mirjalili, S., Faris, H., Aljarah, I., Mafarja, M. and Chen, H., "Harris hawks optimization: Algorithm and applications", Future generation computer systems, vol.97, pp.849-872, 2019.
30. Chu, S.C., Tsai, P.W. and Pan, J.S., "Cat swarm optimization", In proceedings of Pacific Rim international conference on artificial intelligence, pp. 854-858, 2006.